Aparicio-Navarro FJ, Kyriakopoulos KG, Parish DJ.

Empirical Study of Automatic Dataset Labelling.

***In: 9th International Conference for Internet Technology and Secured Transactions (ICITST)*. 2014, London, UK: IEEE.**

DOI link to article:

http://dx.doi.org/10.1109/ICITST.2014.7038840

Date deposited:

11/04/2016

Empirical Study of Automatic Dataset Labelling

Francisco J. Aparicio-Navarro, Konstantinos G. Kyriakopoulos, David J. Parish
School of Electronic, Electrical and System Engineering
Loughborough University
Loughborough, LE11 3TU, UK
e-mail: {elfja2, elkk, d.j.parish}@lboro.ac.uk

Abstract—**Correctly labelled datasets are commonly required. Three particular scenarios are highlighted, which showcase this need. One of these scenarios is when using supervised Intrusion Detection Systems (IDSs). These systems need labelled datasets for their training process. Also, the real nature of analysed datasets must be known when evaluating the efficiency of IDSs detecting intrusions. The third scenario is the use of feature selection that works only if the processed datasets are labelled. In normal conditions, collecting labelled datasets from real communication networks is impossible. In a previous work we developed a novel approach to automatically generate labelled network traffic datasets using an unsupervised anomaly based IDS. The approach was empirically proven to be an efficient unsupervised labelling approach. It was evaluated using a single dataset. This paper extends our previous work by using a greater number of datasets, gathered from a real IEEE 802.11 network testbed. The datasets are comprised of different wireless-specific attacks. This paper also proposes a new and more precise method to calculate the boundary threshold, used in the labelling process.**

Keywords-Automatic Labelling; Network Traffic Labelling; Unsupervised Anomaly IDS; IEEE 802.11 Datasets

I. INTRODUCTION

The need for correctly labelled datasets is generally disregarded in the area of intrusion detection, as it is commonly assumed that real nature of the analysed information is known. As we previously described in [1], there are three specific scenarios in which the need for correctly labelled datasets becomes particularly evident. One of these scenarios is when using supervised Intrusion Detection Systems (IDSs). This type of IDS requires labelled datasets to learn the difference between normal or malicious information. IDSs are commonly classified as supervised and unsupervised detection systems [2]. Unsupervised IDSs learn the difference between normal or malicious information autonomously, whereas supervised IDSs require correctly labelled training datasets to learn the difference. Commonly, training datasets are completely labelled, containing both types of information. If the training datasets is unlabelled, supervised IDSs assume that only non-malicious information is included.

Another of these scenarios in which correctly labelled datasets is needed is when evaluating the efficiency of the IDSs detecting intrusions. IDSs could be evaluated using multiple parameters, such as the amount of resources (CPU, Memory, etc.) the system consumes, or the required time to conduct the detection. Nonetheless, the most important aspect to evaluate IDSs is the number of intrusions that the system correctly identifies. Traditionally, the Detection Rate (DR), False Positive Rate (FPr), and False Negative Rate (FNr) have been the parameters used to evaluate the efficiency of IDSs. These parameters provide quantifiable evidence of how effective the IDSs are at making correct detections. For an IDS to be evaluated in terms of DR, FPr, and FNr, the real nature of the analysed information must be known.

A similar need for correctly labelled datasets arises when feature selection techniques are utilised. Feature selection is used to minimise the number of metrics in a given dataset and to optimise the selection process of the most relevant set of metrics [3]. These techniques play an important role in improving the efficiency of IDSs, producing more accurate results. The use of feature selection is currently inappropriate for unsupervised IDSs, especially if the IDSs perform their detection in real-time. The implementation of automatic feature selection techniques for unsupervised IDSs is still a great challenge in intrusion detection [4]. One of the reasons for this is because feature selection works only if the records in the datasets have been previously labelled [4]. Feature selection requires labelled datasets in order to be able to evaluate the relevance of each metric or combination of metrics.

Unfortunately, collecting labelled datasets from physically deployed networks is highly complicated [5], and in many cases impossible. In normal conditions, real network traffic is not labelled. If researchers controlled the network conditions, or if the network traffic were artificially generated using network simulation software (e.g. OPNET [6]), the instances in the network dataset could be labelled. However, this control of the network environment is not always possible. Even in controlled networks, assuring that the training datasets are correctly labelled or completely free of malicious information is extremely hard [7]. Training datasets are currently generated by implementing a previous off-line forensic analysis.

In our previous work [1] we proposed a novel approach to automatically generate labelled network traffic datasets using the unsupervised anomaly based IDS proposed in [8]. Although it was empirically proven to be an efficient unsupervised labelling approach, this approach was only evaluated using a single dataset. In this paper, our aim is to extend the empirical evaluation of the approach presented in [1], using a greater number of network traffic datasets, comprised of different wireless-specific attacks, gathered from a live operational IEEE 802.11 network testbed. In addition, a new and more advanced method to calculate the boundary threshold than the one presented in [1] is proposed, including an adjustment factor.

The paper is organised as follows. In section II, the most relevant related work is reviewed. In section III, the performance measures used to evaluate the efficiency of IDSs, the description of the different datasets, and the distribution of the different belief values are introduced. The description of the approach for automatically labelling datasets, and the labelling results are presented in section IV. Finally, future work and conclusions are given in section V.

II. Related Work

The need for correctly labelled datasets has been acknowledged multiple times in the literature on IDSs. For instance, the authors of [10] highlight that one of the main requirements for IDS efficiency evaluation is to have access to network traffic data previously labelled as normal or malicious. They also highlight the complexity and time required to implement the labelling process. Another work that highlights the need for correctly labelled datasets is [11]. Similar to [10], the authors of this work [11] highlight the complexity and time required for labelling network traffic data.

There is limited work in this area. One of the few recent papers that target the automatic generation of labelled network traffic datasets is presented in [5]. The authors propose using unsupervised anomaly based IDS to label packet datasets. Their approach is known as a self-training architecture. Similar to our methodology, this work assigns a particular label to each packet based on the beliefs generated by the Dempster-Shafer Theory of Evidence. Using these beliefs, the authors calculate a Reliability Index (RI), and label the packets according to this index. The outcome of the RI is a value in the range [-1, 1] that determines the reliability of the packet labelling. The closer the value to each of the range ends, the higher confidence that the assigned label is correct. The closer to 0, the higher the doubt that the assigned label is correct.

In [5], the authors define a guard region or rejection range. The packets with an RI value that falls in the guard region are rejected. Instead of using a guard region, the methodology that we proposed in [1] defines only a single boundary threshold. One of the main difficulties in [5] is to identify the appropriate limit values for the guard region. On the other hand, one of the difficulties in our work is to identify the appropriate threshold value. One of the disadvantages of the approach in [5] is that the authors need to execute their algorithm multiple times, in order to find the appropriate guard region. An exhaustive search is required. Despite these multiple repetitions, it is not guaranteed that the selected guard region would be appropriate for future data. In our work, the boundary threshold is defined only once for the whole dataset. Therefore, our approach does not require an exhaustive search.

III. Analysis of Belief Difference

A. IDS Performance Measures

The efficiency of IDSs in making correct detections can be evaluated using four well-known parameters. These are True Positive (TP), which represents attack frames correctly classified as malicious; True Negative (TN), which represents non-malicious frames correctly classified as normal; False Positive (FP), which represents non-malicious frames misclassified as malicious; and False Negative (FN), which represents attack frames misclassified as normal. Using these parameters is fundamental to calculating the following performance measures:

- Detection Rate (DR), which is the proportion of malicious frames correctly classified as malicious among all the malicious frames. DR(%) = TP/(FN+TP)
- False Positive Rate (FPr), which is the proportion of non-malicious frames misclassified as malicious among all the frames. FPr(%) = FP/(TP+FP+TN+FN)
- False Negative Rate (FNr), which is the proportion of malicious frames misclassified as normal among all the malicious frames. FNr(%) = FN/(FN+TP)
- Overall Success Rate (OSR), which is the proportion of any frame correctly classified.
  OSR(%) = (TN+TP)/(TP+FP+TN+FN)

B. IEEE 802.11 Network Datasets

The six different network traffic datasets used in the experiments conducted as part of this work have been gathered from a real IEEE 802.11 network testbed. This testbed is similar to the one described in [8], comprising an Access Point (AP), a wireless client accessing various websites on the Internet, a monitoring node and an attacker using Airpwn [9]. These unprocessed datasets are composed of both malicious and non-malicious frames. One of the datasets contains only normal frames. Using a post-gathering forensic analysis, the real nature of the instances in the datasets has been identified.

For each frame in the dataset, the unsupervised IDS provides three levels of belief [1]. These are belief in *Normal*, which indicates how strong the belief is in the hypothesis that the current analysed frame is non-malicious, belief in *Attack*, which indicates how strong the belief is in the hypothesis that the current analysed frame is malicious, and belief in *Uncertainty*, which indicates how doubtful the system is regarding whether the current analysed frame is malicious or non-malicious. In an optimal situation, the detection system should provide high belief in *Normal* and low belief in *Attack* when the analysed frame is non-malicious. Similarly, when the current analysed frame is not from the AP, the detection system should provide high belief in *Attack* and low belief in *Normal*. If the system were not consistent with these criteria, it would be reasonable to assume the result is not accurate.

For each frame, the difference between the belief in *Normal* and *Attack* has been calculated. The average belief difference values for each dataset are plotted in Fig. 1, categorised by the outcome of the IDS (i.e. TP, TN, FP & FN). The bar charts show the correct detection is produced by strong beliefs in the appropriate hypothesis. The frequency histograms of the belief difference for each datasets, representing how the belief results are distributed are shown in Fig. 2. This is the actual difference between the belief in *Normal* and *Attack.* Fig. 2 also shows the boxplots that represent the distribution of the belief difference, using the detection result of the unsupervised anomaly based IDS as the discriminant criteria. This is whether the frames have been correctly classified or not. For each dataset, an in-depth description of the detection results of the IDS and the belief difference is presented in the following section.

1) Normal Traffic Dataset

The first analysed dataset contains only non-malicious network traffic between the AP and the wireless client. In total, 3551 network frames, or instances, compose this dataset. The

unsupervised IDS correctly detects 99.97% of the normal traffic dataset. 3550 instances are correctly classified as normal. Only 1 instance, 0.03% of the dataset, is misclassified. The correct detection is produced by very strong beliefs in *Normal*. In the cases of TN, the average belief in *Normal* is 93.25%, and the average belief in *Attack* is 6.59%. In the cases of FP, the average belief in *Normal* is 38.15%, and the average belief in *Attack* is 61.25%, Fig.1.a. Since this dataset does not contain malicious network data traffic, the unsupervised IDS did not generate any FN or TP alarm when analysing this data.

2) Airpwn Attack Datasets

Three datasets were generated using the Airpwn attack. In the *Attack01* dataset, the attacker replaces the whole content of the website to a custom one. In the *Attack02* dataset, the attacker only replaces the images in the website. Lastly, the *Mix Attack* dataset comprises traces of the two previous types of attack. The Airpwn attack has been previously explained in more detail in [8]. All these datasets contain both malicious and non-malicious network traffic instances.

For the *Attack01* dataset, the IDS correctly classifies 100% of the dataset, 1281 frames. The correct detection is produced by very strong beliefs in the appropriate hypothesis. In the cases of TN, the average belief in *Normal* and *Attack* are 92.39% and 7.41%, respectively. In the cases of TP, the average belief in *Normal* is 7.46%, and the average belief in *Attack* is 92.32%, Fig.1.b. No FP or FN alarms were generated.

For the *Attack02* dataset, the unsupervised IDS correctly classifies 99.98% of the traffic instances. 14413 instances are correctly classified as normal. Only 3 instances, 0.02% of the dataset, are misclassified. In the cases of TP, the average belief in *Normal* and *Attack* are 7.13% and 92.64%, respectively. In the cases of TN, the average belief in *Normal* and *Attack* are 90.2% and 9.57%, respectively. In the cases of FP, the average belief in *Normal* is 42.45%, and the average belief in *Attack* is 57.05%, Fig.1.c. No FN alarm was generated.

Finally, 99.99% of the traffic dataset is correctly classified, for the *Mix Attack* dataset. 12049 instances are correctly classified as normal. Only 1 instance, 0.01% of the dataset, is misclassified. In the cases of TP, the average belief in *Normal* and *Attack* are 6.38% and 93.41%, respectively. In the cases of TN, the average belief in *Normal* and *Attack* are 91.84% and 7.92%, respectively. In the cases of FP, the average belief in *Normal* is 36.92%, and the average belief in *Attack* is 62.57%, Fig.1.d. No FN alarm was generated.

3) Deauthentication Attack Datasets

Using the deauthentication attack, two datasets were generated. Both experiments are implemented using the same attacking tool HostAP [12]. The only difference between both datasets is in the topology of the real IEEE 802.11 testbed in which the datasets were gathered. Although these testbeds are composed of the same number of devices as the one described in [8], for the *DeauthLong* dataset, the attacker is located 10m away from the wireless client, whereas, for the *DeauthShort* dataset, the attacker is located only 1.5m away from the wireless client. This change in the topology produces a small variation in some of the measured metrics.

For the *DeauthLong* dataset, the IDS correctly classifies 98.94% of the dataset, 187 frames. Only 2 instances, 1.06% of the dataset, are misclassified. In the cases of TP, the average belief in *Normal* and *Attack* are 29.85.39% and 69.63%, respectively. In the cases of TN, the average belief in *Normal* is 86.71%, and the average belief in *Attack* is 12.99%. In the cases of FP, the average belief in *Normal* and *Attack* are 35.96% and 63.37%, Fig.1.e. No FN alarms were generated.

For the *DeauthShort* dataset, the unsupervised IDS correctly classifies 96.93% of the traffic instances. 158 instances are correctly classified as normal. Only 5 instances, 3.07% of the dataset, are misclassified. In the cases of TP, the average belief in *Normal* and *Attack* are 30.99% and 68.54%, respectively. In the cases of TN, the average belief in *Normal* and *Attack* are 82.02% and 17.6%, respectively. In the cases of FP, the average belief in *Normal* is 44.27%, and the average belief in *Attack* is 55%, Fig.1.f. No FN alarm was generated.

C. Analysis of the Belief Difference Results

Once the belief difference for all the datasets has been calculated, the mean (μ) and the standard deviation (σ) of the belief difference, containing both malicious and non-malicious frames, are also calculated. These values have been tabulated in Table I, along with the coefficient of Skewness and the Kurtosis values. As explained in Section IV, these two values, μ and σ, will be used in the automatic dataset labelling process.

TABLE I. BELIEFS DIFFERENCE STATISTICAL ANALYSIS

Dataset	Mean (μ)	SD (σ)	Skewness	Kurtosis
Normal	0.866	0.094	-2.712	11.321
Attack01	0.85	0.104	-2.202	6.589
Attack02	0.809	0.116	-2.017	5.641
Mix Attack	0.839	0.105	-2.25	7.664
DeauthLong	0.617	0.228	-0.123	-1.328
DeauthShort	0.522	0.233	0.165	-1.036

As can be seen in all the boxplots in Fig. 2, there is a very clear distinction in the difference values between the correctly classified and the incorrectly classified instances. In order to statistically represent the difference between the correctly and incorrectly classified instances, Table II shows the μ and the σ values of the belief difference for all the datasets. There is an evident statistical disparity in the μ values between the correctly and incorrectly detected frames. Using this difference, if a threshold defining the boundary between the correct and incorrect classifications could be found, misclassified instances could be discarded from the automatically labelled dataset.

TABLE II. BELIEFS DIFFERENCE ANALYSIS – COMPARISON

Dataset	Correct Detection		Incorrect Detection	
	Mean (μ)	*SD (σ)*	*Mean (μ)*	*SD (σ)*
Normal	0.866	0.094	n/a	n/a
Attack01	0.85	0.104	n/a	n/a
Attack02	0.81	0.116	0.146	0.058
Mix Attack	0.839	0.105	0.256	0
DeauthLong	0.621	0.226	0.274	0.258
DeauthShort	0.535	0.225	0.107	0.046

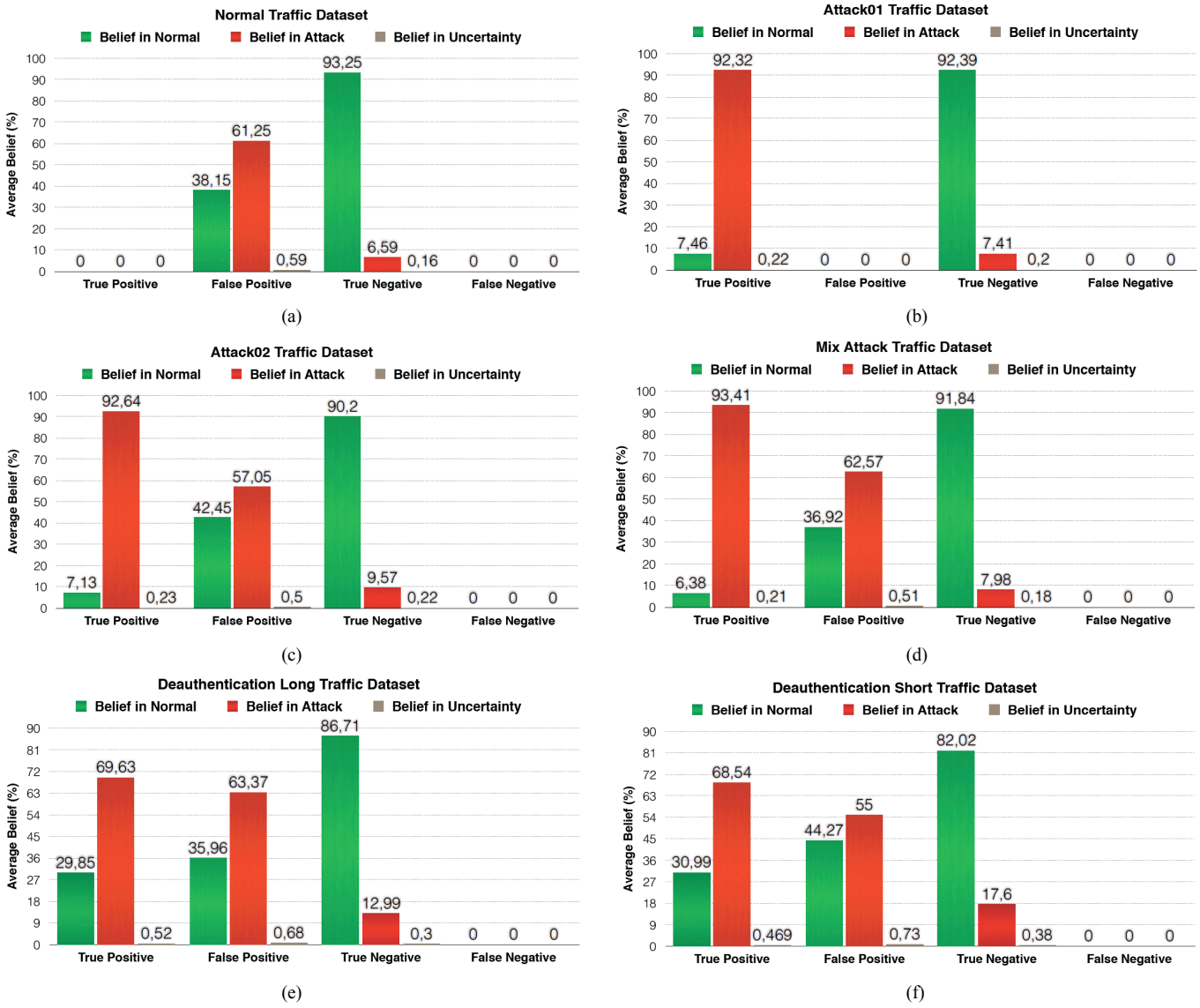

Fig. 1. Bar Charts of the Resulting Set of Metric Frequency: (a) Normal dataset; (b) Attack01 dataset; (c) Attack02 dataset; (d) Mix Attack dataset; (e) Deauthentication Long dataset; (f) Deauthentication Short dataset.

IV. Automatic Dataset Labelling Results

This section describes the approach that we propose to generate the automatically labelled network traffic datasets. This approach was initially introduced in our previous work [1]. The difference between the belief in *Normal* and the belief in *Attack* plays an important role in the correct labelling of the attacks. The system makes use of the statistical disparity in the average belief difference between the correctly and incorrectly detected frames, described in Section III, to generate these labelled datasets. We propose the definition of a boundary threshold that could separate the correctly and incorrectly classified instances. Therefore, the misclassified instances could be discarded from the automatically labelled dataset.

The boundary threshold (γ) is defined by (1), based on the mean and the second standard deviation values. This threshold was previously used in [1]. The threshold described in this paper also includes an adjustment factor (δ) with a value within the range [0, 1]. This adjustment factor is an addition to the method that we previously proposed in [1]. The value of δ allows to fine tune the value of γ, and the integrity of the automatically labelled datasets.

$$\gamma = \mu - (2\sigma) \times \delta \quad (1)$$

The instances with belief difference above γ would be included in the labelled dataset, whereas the instances with belief difference below this threshold would not be included. This approach guarantees that only the cases that evidence strong support to one of the hypotheses are considered correct, and are finally included in the labelled dataset. For instance, the boundary threshold for the *Normal* traffic dataset, using δ = 1, would be γ_{normal} = 0.8664 – (2 × 0.0942) × 1 = 0.678. Any instance where the difference between the belief in *Normal* and the belief in *Attack* is larger than 0.678 will be included in the labelled dataset. In contrast, any instance where the belief difference is smaller than 0.678 will be discarded. After applying (1) to the *Normal* dataset, using δ = 1, 95.92% of the original *Normal* dataset was included in the labelled dataset.

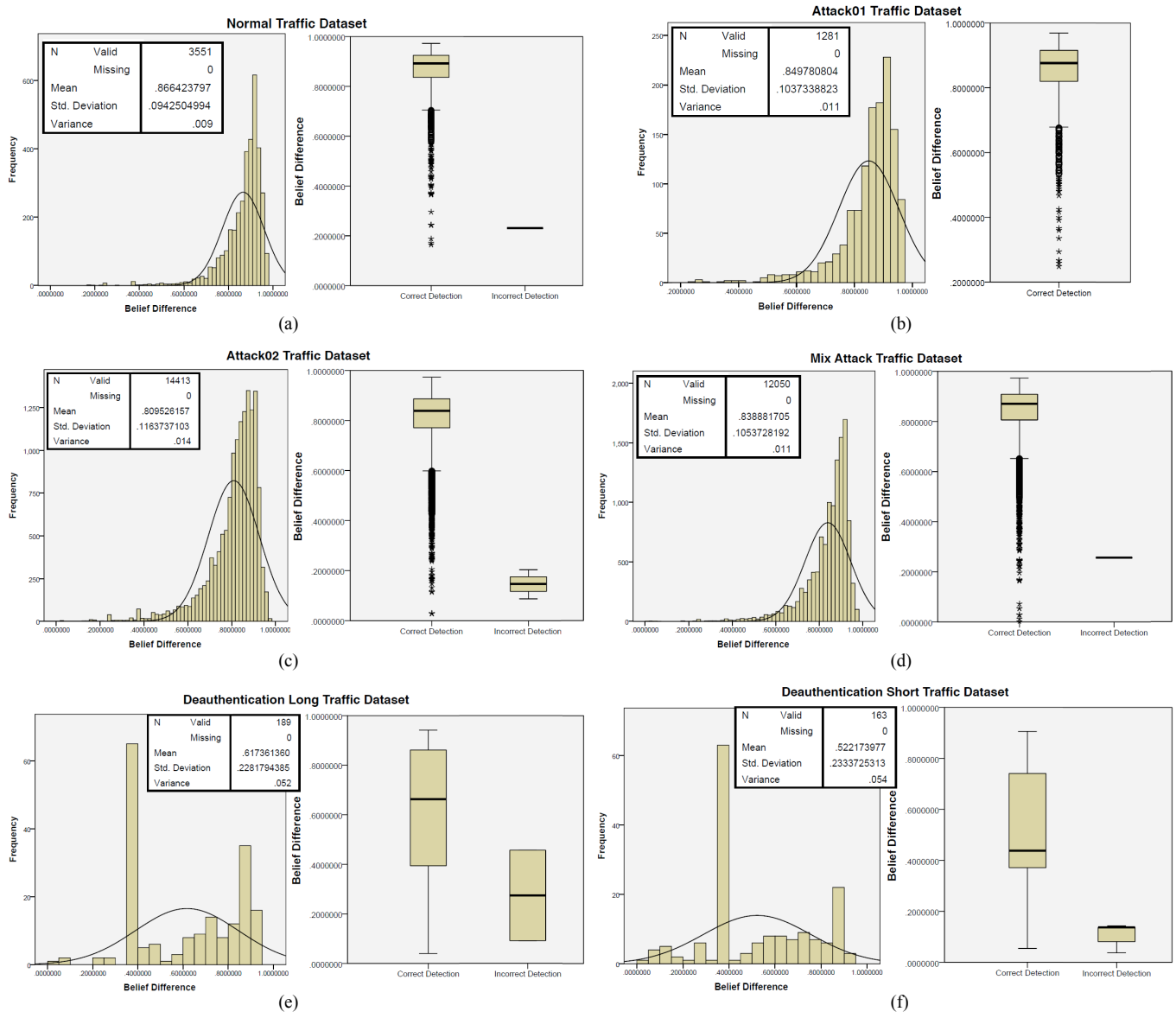

Fig. 2. Histograms and Boxplots - Beliefs Difference of Original Dataset, using Correctly and Incorrectly Classified Instances: (a) Normal traffic dataset; (b) Attack01 dataset; (c) Attack02 dataset; (d) Mix Attack dataset; (e) Deauthentication Long dataset; (f) Deauthentication Short dataset.

In total, 3406 instances compose the new *Normal* dataset, and 100% of the resulting dataset is correctly labelled. All the incorrectly labelled instances have been discarded, along with 146 of the correctly labelled instances. Similarly, none of the new labelled datasets resulting after applying (1) using $\delta = 1$ to the Airpwn datasets contain any misclassified frame. For the *Attack01*, 1212 instances compose the new labelled dataset. This is 94.61% of the original dataset. For the *Attack02* dataset, 94.89% of the original dataset (i.e. 13677 instances) compose the new labelled dataset, and 100% of the frames are correctly labelled. For the *Mix Attack* dataset, 11481 instances, 95.28% of the original dataset compose the new dataset. 570 correctly labelled instances were discarded from the original dataset.

In contrast, after applying (1) to the deauthentication attack datasets, using $\delta = 1$, the resulting labelled datasets still contain misclassified frames. In the case of *DeauthLong* dataset, 1 non-malicious frame erroneously labelled as malicious is included in the resulting dataset. In total, 99.47% of the frames are correctly labelled. In the case of *DeauthShort* dataset, 4 non-malicious frames are included in the resulting dataset, erroneously labelled as malicious. In total, 97.5% of the frames are correctly labelled. These erroneously labelled instances would skew the training process of supervised IDSs, the evaluation of IDSs, or the process of feature selection.

Since the automatic dataset labelling process was not completely accurate using $\delta = 1$, a number of experiments have been conducted with a double purpose. First, to evaluate the quality of the automatic dataset labelling approach, modifying the value of δ, and second, to identify the higher value of δ that would produce completely correct labelling with all the evaluated network datasets. In these experiments, the value of δ has been gradually reduced from 1 to 0. The modification of this value produces a more restrictive γ value, and the resulting

labelled datasets would contain a smaller proportion of the original datasets. However, the smaller the δ value, the higher the probability of including only correctly labelled instances.

For each dataset and each value of δ, the value of γ, the proportion of original datasets included in the labelled datasets, the number of frames that compose the new datasets, and the rate of correctly labelled instances in the resulting datasets have been calculated. Table III shows the results for $\delta = [0.3, 0.4]$.

TABLE III. BOUNDARY THRESHOLD VALUES ($\delta = [0.3, 0.4]$)

Dataset	δ	Boundary Threshold (γ)	Satisfy Threshold (%)	Number of Frames in Dataset	Correctly Labelled (%)
Normal	0.4	0.791	85.81	3047	100
	0.3	0.81	82.09	2915	100
Attack 01	0.4	0.767	88.68	1136	100
	0.3	0.788	85.95	1101	100
Attack 02	0.4	0.716	84.52	12182	100
	0.3	0.74	81.2	11703	100
Mix Attack	0.4	0.755	84.99	10241	100
	0.3	0.776	80.84	9741	100
Deauth Long	0.4	0.435	60.32	113	99.47
	0.3	0.48	56.61	107	100
Deauth Short	0.4	0.335	87.73	143	100
	0.3	0.382	49.08	80	100

All the resulting datasets are 100% correctly labelled when $\delta = 0.3$ is used. This is the higher δ value that makes the system to correctly label all the frames in all of the evaluated datasets. However, as the value of δ decreases, the amount of data included in the labelled datasets also decreases. Using $\delta = 0.3$, the resulting labelled *DeauthLong* dataset contains 56.61% of the original dataset, whereas the resulting labelled *DeauthShort* dataset contains only 49.08% of the original dataset.

The subset of the original datasets discarded during the labelling process could be manually classified implementing an off-line forensic analysis, and added to the automatically labelled dataset if required to ensure a consistent dataset. The manual effort required to do this would be much reduced.

These results highlight a series of inconveniences for the automatic labelling process. First, the selection of the δ value has been done empirically. For an efficient automatic dataset labelling process, the need for a method to autonomously implement the selection of δ should be addressed in future work. Second, there exist a disparity in the value of δ that produces entirely correct labelled datasets, between both deauthentication datasets and the rest of datasets. Whereas using $\delta = 1$, 100% of the instances are correctly labelled for the Normal and Airpwn attack datasets, the two deauthentication attack datasets require $\delta = 0.3$. One of the reasons for this disparity can be the size of the datasets. Whilst the first four datasets contain thousands of instances, both deauthentication datasets contain less than 200 instances. A higher number of instances would make the statistical characteristics of the datasets converge to a more concrete distribution making the automatic dataset labelling process more accurate. There is a clear direct correlation between δ and the number of instances. If the size of the analysed datasets is small, the value of δ needed to autonomously generate 100% correctly labelled datasets should be also small. This correlation between the size of the datasets and the value of δ could be used to propose method to autonomously select the adjustment factor value.

V. CONCLUSIONS

This paper extends the automatic generation of correctly labelled network traffic datasets that we proposed in [1]. This approach uses the outcome beliefs of an unsupervised IDS to label the instances in the datasets. Only the cases that evidence strong support to one of the hypotheses are considered correct. The evaluation of the automatic dataset labelling approach uses six different datasets, gathered from a real IEEE 802.11 network. The method to calculate the threshold γ presented in this paper adds an adjustment factor δ, which allows to fine tune the value of γ, and to increase the integrity of the automatically labelled datasets. The value of δ has been empirically chosen. A new method to independently select this value should be proposed in future work to achieve efficient automatic dataset labelling. Because of the small size of both *Deauth* datasets, the higher δ value that correctly labels the instance of all the evaluated datasets is $\delta = 0.3$. The size of the datasets could be used to propose an automatic method to select the adjustment factor value δ.

ACKNOWLEDGMENT

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/ K014307/1 and the MOD University Research Collaboration in Signal Processing.

REFERENCES

[1] F. J. Aparicio-Navarro, K. G. Kyriakopoulos, and D. J. Parish, "Automatic dataset labelling and feature selection for intrusion detection systems," in Proceedings of the IEEE *Military Communications Conference, MILCOM* 2014, pp. 46-51.

[2] P. Laskov, P. Düssel, C. Schäfer, and K. Rieck, "Learning intrusion detection: supervised or unsupervised?," in Image Analysis and Processing-2005. vol.3617, Springer Berlin Heidelberg, 2005, pp. 50-57.

[3] G. Stein, B. Chen, A. S. Wu, and K. A. Hua, "Decision tree classifier for network intrusion detection with GA-based feature selection," Proceedings of the 43rd annual Southeast regional conference-Volume 2. ACM, 2005, pp. 136-141.

[4] H. Nguyen, K. Franke, and S. Petrovic, "Improving effectiveness of intrusion detection by correlation feature selection," in Proceedings of the IEEE International Conference on Availability, Reliability, and Security, *ARES'10*, 2010, pp. 17-24.

[5] F. Gargiulo, C. Mazzariello, and C. Sansone, "Automatically building datasets of labeled IP traffic traces: A self-training approach," Applied Soft Computing, vol.12, n.6, 2012, pp. 1640-1649.

[6] OPNET Modeler, "Riverbed Technology" http://www.riverbed.com/products/performance-management-control/ (Access Date: 6 Aug, 2014).

[7] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geometric framework for unsupervised anomaly detection," Applications of data mining in computer security. Springer US, 2002, pp. 77-101.

[8] F. J. Aparicio-Navarro, K. G. Kyriakopoulos, and D. J. Parish, "An automatic and self-adaptive multi-layer data fusion system for WiFi attack detection," International Journal of Internet Technology and Secured Transactions, vol.5, n.1, 2013, pp. 42-62.

[9] Airpwn Packet Injection Framework Website Available: http://airpwn.sourceforge.net/Airpwn.html (Access Date: 26 Nov, 2014).

[10] C. A. Catania, and C. García Garino, "Automatic network intrusion detection: Current techniques and open issues," Computers & Electrical Engineering, vol38, n.5, 2012, pp. 1062-1072.

[11] J. J. Davis, and A. J. Clark, "Data preprocessing for anomaly based network intrusion detection: A review," Computers & Security, vol.30, n.6, 2011, pp. 353-375.

[12] J. Malinen, Host AP Website. Available: http://w1.fi/ (Access Date: 26 Nov, 2014).